

Nicholas Newlin

Completed



Original



```
# Nicholas Newlin  
# March 15, 2022
```

```
#the collage is made by changing the source picture 5 different times, and copying a segment of that changed  
#picture onto the canvas
```

```
def collage():
```

```
    canvas = makeEmptyPicture(1000,736,white)
```

```
#this ghost image was taken by me, so I own this image
```

```
    ghost = makePicture(getMediaPath("ghost.jpg"))
```

```
#this creates the first slice (blurred) and copies it onto canvas
```

```
    blurred = blur()
```

```
    copyToCanvas(blurred,canvas,0,200)
```

```
#this creates the second slice (posterize) and copies it onto canvas
```

```
    posterized = posterize()
```

```
    copyToCanvas(posterized,canvas,200,400)
```

```
#this creates the third slice (my own pattern) and copies it onto canvas
```

```
    weirdPatterned = weirdPattern()
```

```
    copyToCanvas(weirdPatterned,canvas,400,600)
```

```
#this creates the fourth slice (grayscale) and copies it onto canvas
```

```
    grayscaled = grayscale()
```

```
    copyToCanvas(grayscaled,canvas,600,800)
```

```
#this creates the fifth slice (edge detect) and copies it onto canvas
```

```
    edgeDetected = edgeDetect()
```

```
    copyToCanvas(edgeDetected,canvas,800,1000)
```

```

signature(canvas)
explore(canvas)

def signature(target):
    signature = makePicture(getMediaPath("sig.jpg"))
    canvasBefore = duplicatePicture(target)
    #this code copies the signature to canvas
    targetX = 0
    for sourceX in range(0,getWidth(signature)):
        targetY = 0
        for sourceY in range(0,getHeight(signature)):
            setColor(getPixel(target,targetX,targetY),getColor(getPixel(signature,sourceX,sourceY)))
            targetY = targetY + 1
        targetX = targetX + 1
    #this code chromakeys the white out and replaces the black writing with red
    for x in range(0,209):
        for y in range(0,31):
            if distance(getColor(getPixel(target,x,y)),makeColor(255,255,255)) < 100:
                setColor(getPixel(target,x,y),getColor(getPixel(canvasBefore,x,y)))
            if distance(getColor(getPixel(target,x,y)),makeColor(0,0,0)) < 100:
                setColor(getPixel(target,x,y),black)
    return (target)

#this function is used several times in the main function to copy a segment and paste it onto canvas
def copyToCanvas(source,target,startX,endX):
    targetX = startX
    for sourceX in range(startX,endX):
        targetY = 0
        for sourceY in range(0,getHeight(source)):
            setColor(getPixel(target,targetX,targetY),getColor(getPixel(source,sourceX,sourceY)))
            targetY = targetY + 1
        targetX = targetX + 1
    return(target)

#this function is used to create slice 4
def grayscale():
    picture = makePicture(getMediaPath("ghost.jpg"))
    for x in range(600,800):
        for y in range(0,getHeight(picture)):
            pixel = getPixel(picture,x,y)
            intensity = (getRed(pixel)+getGreen(pixel)+getBlue(pixel))/3
            setColor(pixel,makeColor(intensity,intensity,intensity))
    return (picture)

```

```

#this function is used to create slice 2
def posterize():
    picture = makePicture(getMediaPath("ghost.jpg"))
    for x in range(200,400):
        for y in range(0,getHeight(picture)):
            px = getPixel(picture,x,y)
            if getRed(px) < 64:
                setRed(px, 31)
            if 63 < getRed(px) < 128:
                setRed(px, 95)
            if 127 < getRed(px) < 192:
                setRed(px, 159)
            if 191 < getRed(px) < 256:
                setRed(px, 223)

            if getGreen(px) < 64:
                setGreen(px, 31)
            if 63 < getGreen(px) < 128:
                setGreen(px, 95)
            if 127 < getGreen(px) < 192:
                setGreen(px, 159)
            if 191 < getGreen(px) < 256:
                setGreen(px, 223)

            if getBlue(px) < 64:
                setBlue(px, 31)
            if 63 < getBlue(px) < 128:
                setBlue(px, 95)
            if 127 < getBlue(px) < 192:
                setBlue(px, 159)
            if 191 < getBlue(px) < 256:
                setBlue(px, 223)
    return (picture)

```

#this function is used to create slice 3 and was created by me. I messed with the numbers in the range #statements to make a cool pattern

```

def weirdPattern():
    secondPic = makePicture(getMediaPath("ghost.jpg"))
    picture = makeEmptyPicture(getWidth(secondPic),getHeight(secondPic),lightGray)
    for x in range (400, 600, 2):
        for y in range (0, getHeight(picture), 5):
            sourcePixelColor = getColor(getPixel(picture,x,y))
            setColor(getPixel(secondPic,x,y),sourcePixelColor)
    for x in range (400, 600, 5):
        for y in range (0, getHeight(picture), 2):

```

```

    sourcePixelColor = getColor(getPixel (picture, x, y))
    setColor (getPixel (secondPic, x, y), sourcePixelColor)
return (secondPic)

#this function is used to create slice 1
def blur():
    picture = makePicture(getMediaPath("ghost.jpg"))
    blurredPicture = duplicatePicture(picture)
    for x in range(1, getWidth(picture)-1):
        for y in range(1, getHeight(picture)-1):
            top = getPixel (picture, x, y-1)
            left = getPixel (picture, x-1, y)
            bottom = getPixel (picture, x, y+1)
            right = getPixel (picture, x+1, y)
            center = getPixel (blurredPicture, x, y)
            newRed=(getRed(top)+ getRed(left) + getRed(bottom) + getRed(right) + getRed(center))/5
            newGreen=(getGreen(top) + getGreen(left) + getGreen(bottom)+ getGreen(right)+getGreen(center))/5
            newBlue=(getBlue(top) + getBlue(left) + getBlue(bottom) + getBlue(right)+ getBlue(center))/5
            setColor (center, makeColor(newRed, newGreen, newBlue))
    return (blurredPicture)

#this function is used to create slice 5
def edgeDetect():
    picture = makePicture(getMediaPath("ghost.jpg"))
    for x in range(800,1000):
        for y in range(0,getHeight(picture)):
            px = getPixel (picture, x, y)
            if y < getHeight (picture)-1 and x < getWidth(picture)-1:
                botrt = getPixel (picture, x+1, y+1)
                thislum = luminance (px)
                brlum = luminance (botrt)
                if abs(brlum-thislum) > 10:
                    setColor (px, gray)
                if abs(brlum-thislum) <= 10:
                    setColor (px, lightGray)
    return (picture)

#this luminance function is only used in the edgeDetect function
def luminance(pixel):
    r = getRed(pixel)
    g = getGreen(pixel)
    b = getBlue(pixel)
    return (r+g+b)/3

```